

ÍNDICE

1. INTRODUCCIÓN A CSS.....	2
1.1. EXTENSIONES CSS VISUAL STUDIO CODE.....	2
1.2. HOJAS DE ESTILO EN CASCADA.....	3
1.3. INSERCIÓN DE ESTILOS.....	3
1.4. PSEUDOCASES.....	4
1.5. PSEUDOSELECTORES DE ESTADO.....	4
1.6. PESUDOELEMENTOS.....	4
2. APLICACIÓN DE ESTILOS BÁSICOS.....	5
2.1. TEXTO.....	5
2.2. LISTAS.....	7
2.3. TABLAS.....	7
2.4. CAJAS: MÁRGENES, BORDES, PADDING.....	8
2.5. COLORES, TRANSPARENCIA Y FONDOS.....	10
2.6. FONDOS MÚLTIPLES.....	11
2.7. FORMULARIO BÁSICO.....	11
2.8. POSICIONAMIENTO.....	11
2.9. FORMULARIO AVANZADO.....	12
2.10. MENÚS.....	12
2.11. EJEMPLO RESUMEN.....	12
3. EFECTOS.....	13
3.1. SOMBRAS.....	13
3.2. GRADIENTES.....	13
3.3. TRANSICIONES.....	13
3.4. TRANSFORMACIONES.....	13
3.5. ANIMACIONES.....	13
4. BUENAS PRÁCTICAS CSS.....	14
5. PREFIJOS CSS DE LOS NAVEGADORES.....	16

LINKS DE AYUDA

<https://docs.emmet.io/cheat-sheet/>

<https://lenguajecss.com/css/>

<https://www.w3schools.com/css/>

<https://www.w3schools.com/bootstrap5/>

<https://www.w3schools.com/sass/>

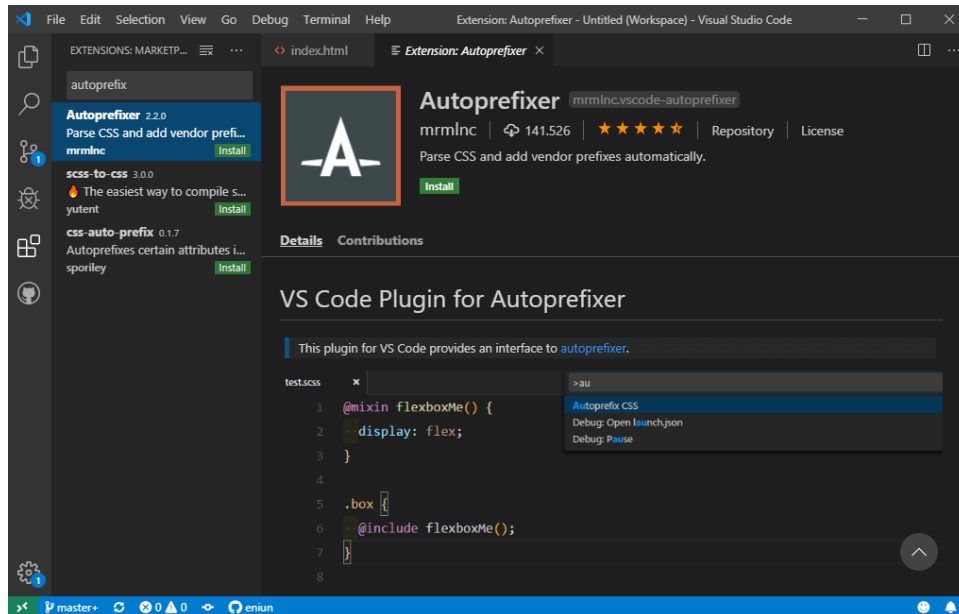
<https://www.eniun.com/que-es-css-cascading-style-sheets/>

1. INTRODUCCIÓN A CSS

1.1. EXTENSIONES CSS VISUAL STUDIO CODE

AUTOPREFIXER

Para ahorrar tiempo y facilitarnos la tarea de incluir los prefijos de las propiedades CSS que todavía no son estables podemos hacer uso de la **extensión «Autoprefixer» en Visual Studio Code**.



Instalar la extensión la **extensión «Autoprefixer**

En view /command palette (o Ctrl+shift+p) escribimos **«autoprefixer run»** para añadir todos los prefijos necesarios en una hoja de estilos.

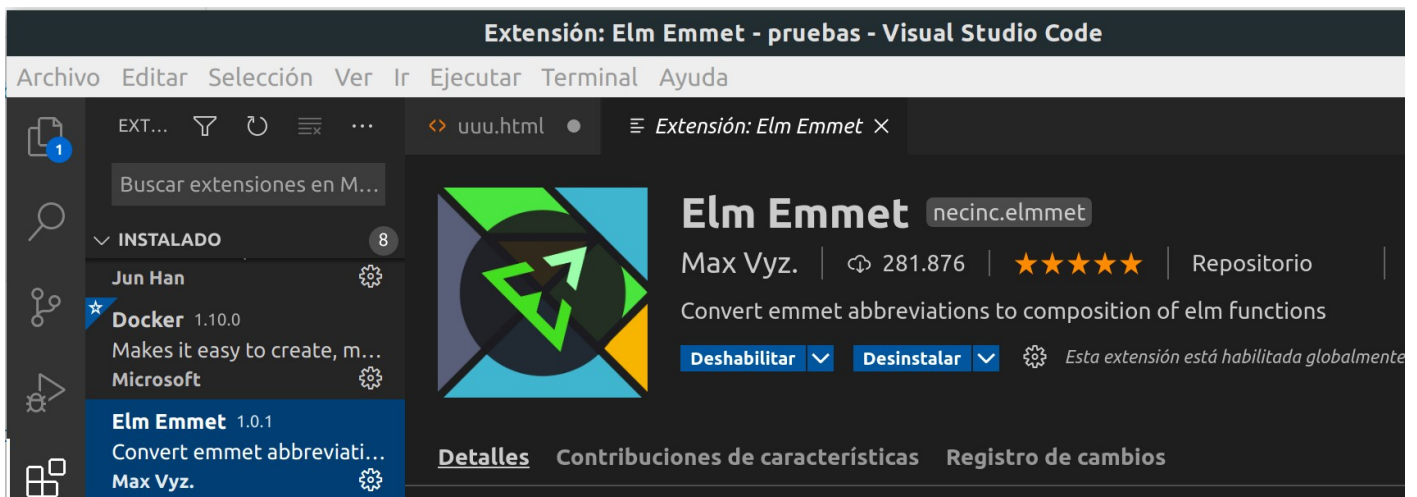
Puedes probar que funciona añadiendo esta propiedad en un estilo CSS y viendo cómo se insertan sus correspondientes prefijos:

user-select: none; /* controla si el usuario puede seleccionar el texto */

Desde los ajustes de la extensión se puede indicar que se busque sin faltan prefijos cada vez que se guarda el documento.

EMMET

Permite usar abreviaturas para generar código html y css



1.2. HOJAS DE ESTILO EN CASCADA

CSS (Cascading Style Sheets) es un **estándar W3C** que define la presentación de documentos Web. En www.w3.org/Style/CSS encontrarás información **actualizada sobre las actividades** del W3C. CSS se mantiene **instrucciones separadas del contenido** del documento HTML.

1.3. INSERCIÓN DE ESTILOS

1. ETIQUETA STYLE: [Ejemplo1](#) || <https://codepen.io/profe-ana/pen/mdLGJQo>
2. ETIQUETA LINK Y FICHERO EXTERNO: [Ejemplo2](#) || <https://codepen.io/profe-ana/pen/OJZoJPV>

SELECTORES

SELECTOR	DESCRIPCIÓN
*	Todos los elementos
etiqueta	Etiquetas indicadas
.class	Elementos con .class
#id	Elemento con id #id

Selector	Descripción
sel1 sel2	sel2 dentro de sel1
.class1.class2	Elementos con dos clases
sel1.class1	sel1 con class1
sel1, sel2	Selectores separados por comas
sel1 > sel2	sel2 hijo de sel1
sel1 + sel2	sel2 hermano de sel1 (su padre es el mismo)

<https://codepen.io/Eniun/pen/gNdZXL>

SELECTORES DE ATRIBUTOS

SELECTOR	SIGNIFICADO	EJEMPLO
Etiqueta [atributo= ^valor]	comience por valor	a[href=^https]
Etiqueta [atributo= *valor]	contenga ese valor	img[alt=*logo:]
Etiqueta [atributo= \$valor]	acaben en ese valor	img [src=\$png]

<https://codepen.io/profe-ana/pen/XWqPWdE>

1.4. PSEUDOCCLASES

HIJOS	DESCRIPCIÓN
:first-child	Primer hijo
:last-child	Último hijo
:nth-child(n)	Enésimo hijo
:nth-last-child(n)	Enésimo hijo contando desde el último
:only-child	Hijos únicos
:empty	Sin hijos

firstchild-lastchild: <https://codepen.io/profe-ana/pen/WNJgNXj>

nth-child odd y even <https://codepen.io/profe-ana/pen/RwyYNmQ>

HERMANOS	DESCRIPCIÓN
:first-of-type	Primer hermano de su tipo
:last-of-type	Último hermano de su tipo
:only-of-type	Únicos hermanos de su tipo
:nth-of-type(n)	Enésimo hermano de su tipo
:nth-last-of-type(n)	Enésimo hermano de su tipo contando desde el último

<https://codepen.io/profe-ana/pen/NPxjOae?editors=1100>

1.5. PSEUDOSELECTORES DE ESTADO

Pseudo-clase	Descripción
:link	No visitado
:visited	Visitado
:hover	Apuntador pasa por encima
:active	Se activa cuando se pulsa
:focus	Se activa cuando tiene el foco

<https://codepen.io/Eniun/pen/PoqBoKY>

<https://codepen.io/profe-ana/pen/MWGqWeO>

not-selector <https://codepen.io/profe-ana/pen/dyeqyJx>

1.6. PESUDOELEMENTOS

Pseudo-elemento	Descripción
::first-line	Primera línea de texto
::first-letter	Primera letra de la primera línea de texto
::before	Añade contenido al principio
::after	Añade contenido al final
::selection	Coge la porción del texto que se está seleccionando por el usuario

<https://codepen.io/Eniun/pen/QWbBWaK>

2. APLICACIÓN DE ESTILOS BÁSICOS

2.1. TEXTO

TEXTO Y EJEMPLO	DESCRIPCIÓN	VALORES
text-align https://codepen.io/Eniun/pen/ewQebg	Alineación horizontal	left right center justify
vertical-align https://codepen.io/Eniun/pen/ZEBWxve https://codepen.io/Eniun/pen/oNYxqdP	Alineación vertical	top middle bottom baseline sub super valor
text-indent https://codepen.io/Eniun/pen/rEQYGM	Desplazamiento de primera línea	longitud porcentaje
Line-height https://codepen.io/Eniun/pen/xoQpOz	Interlineado	longitud porcentaje
text-decoration https://codepen.io/Eniun/pen/QXJOYb	Efectos de subrayado	none underline overline line-through
letter-spacing https://codepen.io/Eniun/pen/ewQyYR	Espacio entre caracteres	normal longitud
word-spacing https://codepen.io/Eniun/pen/RzqxPO	Espacio entre palabras	normal longitud
text-transform https://codepen.io/Eniun/pen/EBOoPQ	Transformación a mayúsculas / minúsculas	capitalize uppercase lowercase none
COLUMNAS https://codepen.io/profe-ana/pen/rNvZVbG		
column-width:	column-count:	column-gap:
		column-rule:

FUENTE Y EJEMPLO	DESCRIPCIÓN	VALORES
font-family https://codepen.io/Eniun/pen/bPQypr	Familias	nombre-familia *
font-size https://codepen.io/Eniun/pen/xoQNMw	Tamaño	xx-small x-small small medium large x-large xx-large larger smaller longitud porcentaje
font-weight https://codepen.io/Eniun/pen/WqYBOM	Anchura Normal = 400, Negrita = 700	normal bold bolder lighter 100 200 300 400 500 600 700 800 900
font-style https://codepen.io/Eniun/pen/KjrLRG	Estilo	normal italic oblique
font-variant https://codepen.io/Eniun/pen/pXQmOq	Convierte a mayúsculas	normal small-caps

EJEMPLO FUENTES <https://lenguajecss.com/css/fuentes-y-tipografias/tipografias-externas/>

Los archivos de fuentes deben de estar al mismo nivel que la hoja de estilos.

FORMATOS FUENTES

Actualmente no hay un formato de fuente único para todos los navegadores.

Es necesario incluir las fuentes en todos los formatos para que sean visibles en el mayor número de navegadores. Los formatos más destacados son los siguientes:

Formato	Descripción
EOT	Embedded OpenType, Explorer
TTF	TrueType Font, IOS
WOFF	Web Open Font Format, Chrome
WOFF2	Mejora de WOFF
SVG	Scalar Vector Graphics

@font-face en CSS

Permite descargar una fuente o tipografía, cargarla en el navegador y utilizarla en nuestras páginas.

Para incluir la fuente Open Sans:

```
@font-face {  
  font-family: 'Open Sans';  
  src: local('Open Sans'),  
       url(opensans.woff2) format('woff2'),  
       url(opensans.woff) format('woff'),  
       url(opensans.ttf) format('truetype'),  
       url(opensans.otf) format('opentype'),  
       url(opensans.eot) format('embedded-opentype');  
}
```

REPOSITORIOS FUENTES

Uno de los repositorios proveedores de tipografias más utilizados es [Google Fonts](https://fonts.google.com/) (fonts.google.com).

Los motivos de su uso son los siguientes:

- **Gratuito:** dispone de un amplio catálogo de fuentes gratuitas.
- **Cómodo:** su interfaz es de fácil uso.
- **Rápido:** utiliza un CDN que proporciona ventajas en cuanto a la velocidad.

<https://codepen.io/Eniun/pen/WNNGNKb>

2.2. LISTAS

PROPIEDAD Y EJEMPLO	DESCRIPCIÓN	VALORES
List-style https://codepen.io/Eniun/pen/QXzLXJ	varios estilos de la lista en una sola propiedad	list-style- type list-style- position list-style- image
List-style-type https://codepen.io/Eniun/pen/ewQwLd	Estilo de marcador	disc circle square decimal decimal-leading-zero lower-roman upper-roman lower-greek lower-latin upper-latin armenian georgian lower-alpha upper-alpha none
List-style-image https://codepen.io/Eniun/pen/rEQEEy https://codepen.io/profe-ana/pen/xxjaGxg	Imagen de marcador	url(«...») none
List-style-position https://codepen.io/Eniun/	Posición del marcador	inside outside

EJEMPLOS DE BOTONERA HORIZONTAL CON LISTAS

<https://codepen.io/Eniun/pen/JjjGZQp>

<https://codepen.io/Eniun/pen/RwZWNEV>

<https://codepen.io/Eniun/pen/BadoyMr>

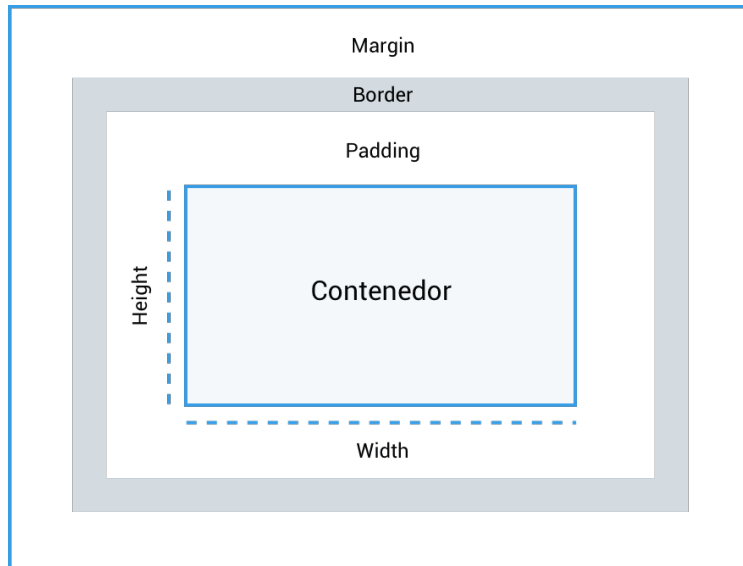
<https://codepen.io/profe-ana/pen/bGMxNXz>

2.3. TABLAS

Propiedad y Ejemplo	Descripción	Valores
caption-side https://codepen.io/Eniun/pen/ZdVWpM	Posición del título en la tabla	top bottom
table-layout https://codepen.io/Eniun/pen/QXzNxx	Formato de las celdas, filas y columnas	auto fixed
border-collapse https://codepen.io/Eniun/pen/qzLNNq	Selección del modelo de los bordes	collapse separate
border-spacing https://codepen.io/Eniun/pen/bPOpyK	Espaciado entre bordes de celdas adyacentes	longitud
empty-cells https://codepen.io/Eniun/pen/agPZWm	Visibilidad de los bordes de celdas sin contenido	show hide

EJEMPLO TABLA: <https://codepen.io/profe-ana/pen/MWGqwYw>

2.4. CAJAS: MÁRGENES, BORDES, PADDING



Valores de padding y margin:(sentido horario)

- **Un valor:**se aplica a los **4 lados**.
- **Dos valores:**el primer valor se aplica a **arriba y abajo**, el segundo valor se aplica a **derecha e izquierda**.
- **Tres valores:**el primer valor se aplica a **arriba**, el segundo valor a **derecha e izquierda** y el tercer valor se aplica a **abajo** del elemento.
- **Cuatro valores:**el primer valor se aplica a **arriba**, el segundo valor se aplica a la **derecha**, el tercer valor se aplica a **abajo** y el cuarto valor se aplica a la **izquierda**.

padding	Descripción	Valores
padding	Tamaño del relleno	longitud porcentaje {1,4}
padding-top -right -bottom -left	Tamaño del relleno superior, derecho, inferior e izquierdo	longitud porcentaje

<https://codepen.io/Eniun/pen/orVxjQ>

margin	Descripción	Valores
margin	Ancho de los márgenes	longitud porcentaje auto {1,4}
margin-top -right -bottom -left	Tamaño del margen superior, derecho, inferior e izquierdo	longitud porcentaje auto

<https://codepen.io/Eniun/pen/RzdrjY>

border	Descripción	Valores
border	Ancho, estilo y color para los bordes	border-width border-style border-color
border-top border-bottom border-right border-left	Ancho, estilo y color para el borde superior, derecho, inferior e izquierdo	border-top-width border-top-style border-top-color
border-width	Anchura del borde	thin medium thick longitud {1,4}
border-top-width border-right-width border-bottom-width border-left-width	Anchura borde superior, derecho, inferior o izquierdo	thin medium thick longitud
border-color	Color del borde (reducida)	color transparent {1,4}
border-top-color border-bottom-color border-right-color border-left-color	Color del borde superior, derecho, inferior e izquierdo	color transparent
border-style	Estilo del borde (reducida)	none hidden dotted dashed solid double groove ridge inset outset {1,4}
border-top-style border-bottom-style border-right-style border-left-style	Estilo del borde superior, derecho, inferior e izquierdo	none hidden dotted dashed solid double groove ridge inset outset
border-radius	Curvatura del borde	longitud porcentaje {1,4}

EJEMPLOS BORDER

<https://codepen.io/Eniun/pen/JQzmPL>

<https://codepen.io/Eniun/pen/gOLxJdB>

BOX-SIZING

Por defecto el **ancho y alto de una caja** es aplicado sólo al **contenido**.

Se debe ajustar el ancho y alto para incluir un borde o padding.

Es útil para **redimensionar el contenido**.

Los valores de box-sizing son:

- **content-box:** es el valor por defecto
 - Si se define el ancho de 100 px,
 - El contenido tendrá 100 px de ancho, y el ancho del borde o padding se **añade** al ancho
- **border-box:**
 - Si se define un ancho de 100 px.
 - Esos 100 pixeles **incluyen** borde o padding que se añada, y la caja de contenido se **encogerá** para absorber ese ancho extra.
 - **Ejemplo:** <https://codepen.io/Eniun/pen/mddBGWV>

PROPIEDAD RESIZE

Por defecto `<textarea>` permiten cambiar el tamaño por el usuario.

Podemos anular este comportamiento con «**resize:none**».

2.5. COLORES, TRANSPARENCIA Y FONDOS

FORMATOS COLOR

- **RGB** (Red, Green, Blue)
- **RGBA** (Red, Green, Blue, Alpha)
- **HSL** (Hue, Saturation, Lightness)
- **HSLA** (Hue, Saturation, Lightness, Alpha)

COLOR RGB

Para definir color indicar:

- **Nombre** del color en inglés: **pink**
- **RGB número** del color en **hexadecimal**: **#FF00EE**
- **RGB número** expresado en **decimal**: **rgb (34,230,124)**

Para añadir **transparencia** se añade un 4º valor entre 0 y 1: **rgba(20,20,20,0.80)**

LISTADO colores [140 nombres de colores](#)

EJEMPLO COLOR: <https://codepen.io/Eniun/pen/yLoNx Bj>

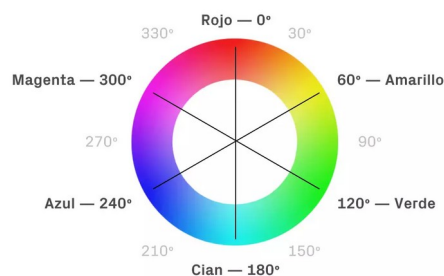
EJEMPLO DE COLOR CON TRANSPARENCIA: <https://codepen.io/profe-ana/pen/MWGqwy a>

COLOR HSL (COLOR, SATURACIÓN Y BRILLO)

EJ: **hls(306,10%,60%)** /* color 306, saturación 10% y brillo 60%*/

La ventaja de HSL es que se puede saber el color de manera intuitiva, ya que se basa en una rueda de color de 360º, donde cada color se sitúa en los siguientes puntos:

- **rojo:** 0%
- **amarillo:** 60%
- **verde:** 120%
- **cyan:** 180%
- **azul:** 240%
- **magenta:** 300%



También podemos aplicar transparencia mediante **hsla**, añadiendo un cuarto valor Alpha (de 0 a 1)

EJEMPLO COLOR HSL: <https://codepen.io/profe-ana/pen/XWqPbNN>

PROPIEDADES COLOR Y FONDO

PROPIEDAD	DESCRIPCIÓN	VALORES
color	Color del texto	RGB HSL HEX nombre del color RGBA HSLA
background-color	Color de fondo	RGB HSL HEX nombre del color RGBA HSLA
background-image	Imagen de fondo	url(«...») none
background-repeat	Repetición de imagen de fondo	repeat repeat-x repeat-y no-repeat
background-attachment	Desplazamiento de imagen de fondo	scroll fixed
background-position	Posición de imagen de fondo	percentage length left center right
background-size	Tamaño de imagen de fondo	valor
Opacity	Transparencia	[0 – 1] (0 → totalmente transparente)

EJEMPLOS COLOR Y FONDO:

<https://codepen.io/Eniun/pen/vqzbor>

<https://codepen.io/Eniun/pen/xxRLPpa>

<https://codepen.io/Eniun/pen/YzpxEvo>

2.6. FONDOS MÚLTIPLES

<https://codepen.io/profe-ana/pen/WbrjBxa>

2.7. FORMULARIO BÁSICO

Ejemplo formularios en tabla: <https://codepen.io/profe-ana/pen/wvjEaaN>

Formulario sencillo: <https://codepen.io/Eniun/pen/xxxEZwz>

2.8. POSICIONAMIENTO

Propiedad	Descripción	Valores
top right bottom left	Desplazamiento respecto al borde superior, derecho, inferior o izquierdo	longitud porcentaje auto
display	Comportamiento del contenedor	inline block inline-block none
box-sizing	Control de bordes y relleno en el comportamiento del contenedor	content-box border-box
position	Esquema de posicionamiento	static relative absolute fixed sticky
float	Posicionamiento flotante	left right none
clear	Control de cajas adyacentes a las float	none left right both
z-index	Nivel de la capa	auto número entero
visibility	Muestra u oculta un elemento	visible hidden

DISPLAY: <https://codepen.io/Eniun/pen/VJRVJb>

POSITION

- **static:** los elementos se posicionan de acuerdo al flujo normal de la página. Es la **posición natural de los elementos**. No son afectados por **top, bottom, left y right**.
<https://codepen.io/Eniun/pen/agxQJM>
- **relative:** los elementos se posicionan de acuerdo al flujo normal de la página. Si son afectados por **top, bottom, left, right**
<https://codepen.io/Eniun/pen/vqMQjK>
- **fixed:** los elementos se posicionan de forma relativa a la ventana del navegador. Su posición permanece **fija** aunque se desplace la ventana.
<https://codepen.io/Eniun/pen/zVXyGY>
<https://codepen.io/profe-ana/pen/PoedqWX>
- **absolute:** los elementos se posicionan de forma relativa al **padre** que tenga una posición distinta a static.
<https://codepen.io/Eniun/pen/PrgxMO>
- **sticky:** los elementos son posicionados de forma relativa hasta que su bloque contenedor alcanza un límite establecido.
 - Se usa cuando queremos que un elemento tenga una posición relativa hasta un punto y que luego cambie a una posición fija.
 - Si tenemos un banner flotante y nos interesa que al hacer scroll se mantenga fijo y visible.
<https://codepen.io/Eniun/pen/porjLM>

FLOAT

- Valores: **left** | **right** | **none**
- El elemento sale del flujo normal y aparece posicionado a la izquierda o a la derecha de su contenedor, donde el resto de elementos de la página se posicionarán alrededor.

<https://codepen.io/Eniun/pen/wvvrEed>

<https://codepen.io/profe-ana/pen/XWqPbVR>

CLEAR

- Valores: **none** | **left** | **right** | **both**
- Establece si un elemento debe estar al lado de los elementos flotantes que lo preceden o si debe situarse bajo de ellos.
- Se suele utilizar para **restaurar el flujo normal del documento** y así los elementos dejan de flotar.

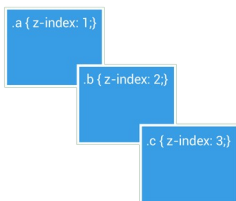
<https://codepen.io/Eniun/pen/ZEWbJJV>

CENTRAR: MARGIN: 0 AUTO 0 AUTO || <https://codepen.io/profe-ana/pen/YzLOXeO>

Z-INDEX

- Organiza cada elemento del contenido para superposiciones.
- Se indica el orden mediante número.
- Primero se visualiza el que tiene el número mayor.

Valores: **auto** | **número**



<https://codepen.io/profe-ana/pen/wBMdNdE>

VISIBILITY

Indica si un elemento es visible u oculto (ocupando el mismo espacio).

Valores: **visible** | **hidden**

<https://codepen.io/profe-ana/pen/PoeBJKv>

2.9. FORMULARIO AVANZADO

<https://codepen.io/profe-ana/pen/zYjJOgR>

2.10. MENÚS

GENERACIÓN DE MENÚS DESDE LAS WEB:

<https://css3menu.com/>

<https://www.easycssmenu.com/>

EJEMPLO MENÚ: <https://codepen.io/profe-ana/pen/vELmPBa>

2.11. EJEMPLO RESUMEN

BLOG CON ASIDE: <https://codepen.io/profe-ana/pen/qEbmvNx>

3. EFECTOS

3.1. SOMBRAS

PROPIEDADES	VALORES EN EL SIGUIENTE ORDEN
text-shadow	posx : sombra que se aplica en el eje horizontal. posy : sombra que se proyecta en el eje vertical. desenfoque o blur : cantidad de sombra borrosa. tamaño : tamaño de la sombra hacia los lados. color : color de la sombra. inset : posiciona la sombra dentro del marco.
box-shadow	

Importante:

- Para aplicar sombras arriba y a la izquierda: utilizar valores negativos.
- Para aplicar doble sombra : dos valores separados con coma.
- Para este tipo de propiedades debemos incluir los **prefijos para navegadores**.

<https://codepen.io/profe-ana/pen/KKRBXeg>

<https://codepen.io/Eniun/pen/GRRrdYL>

Generador de sombras: <https://htmlcheatsheet.com/css/>

3.2. GRADIENTES

<https://codepen.io/profe-ana/pen/rNvrdRP>

<https://codepen.io/profe-ana/pen/vYjajBy>

<https://codepen.io/Eniun/pen/abNWrXV>

Generador de gradientes: <https://htmlcheatsheet.com/css/>

3.3. TRANSICIONES

<https://codepen.io/profe-ana/pen/wvjxjeX>

<https://codepen.io/Eniun/pen/GRRmpKo> |

<https://codepen.io/monchito/pen/nZqoNB>

3.4. TRANSFORMACIONES

<https://codepen.io/profe-ana/pen/gOzjzGv>

<https://codepen.io/Eniun/pen/mddmVWX>

3.5. ANIMACIONES

6 Ejemplo animaciones || <https://codepen.io/profe-ana/pen/KKRBRyE>

4. BUENAS PRÁCTICAS CSS

Organizar la estructura de arriba hacia abajo

Con el fin de encontrar rápidamente cualquier parte de código, es recomendable organizar los estilos de arriba hacia abajo.

```
/****** generic classes******/
styles goes here...
/****** header ******/
styles goes here...
/****** nav menu ******/
styles goes here...
/****** main content ******/
styles goes here...
/****** footer ******/
```

Nombrar correctamente los selectores

Para conseguir más claridad en el código y soporte en todos los navegadores conviene no comenzar el nombre de los selectores con mayúsculas, números ni caracteres especiales.

Evitar: #1div, :-div, DivContent

Mejor utilizar: #div1, .div, divContent

Separar las palabras mediante guiones o mediante mayúsculas

Es recomendable escoger una única manera de escribir el nombre de los selectores. Además, es mejor no usar guiones bajos “_” u otros caracteres especiales ya que algunos navegadores no los soportan. Conviene seguir alguna de las siguientes opciones:

```
/* Opción 1: Palabras separadas por mayúsculas */
navMenu { padding: 2em; border: 2px solid green; }

/* Opción 2: Palabras separadas por guiones */
nav-menu { padding: 2em; border: 2px solid green; }
```

Legibilidad

Adopta una única forma de escribir tu código para que sea más fácil de mantener y de encontrar cualquier elemento.

```
/* Opción 1: Estilos en una línea */
.nav-menu { padding: 2em; border: 2px solid green; }

/* Opción 2: Cada estilo en una línea */
.nav-menu {
    padding: 2em;
    border: 2px solid green;
}
```

Combinar elementos

Cuando varios elementos disponen de las mismas propiedades es recomendable compartirlas en vez de volver a repetir el código. Para ello podemos utilizar [selectores combinados](#), tal y como hemos visto anteriormente.

```
h1, h2, h3 { font-family: Arial; font-weight: 700; }
```

Utilizar selectores descendientes

Es conveniente utilizar [selectores descendientes](#) siempre que sea posible antes de crear un selector clase o un selector identificador. De esta forma, el código estará más limpio, dispondrá de menos selectores clase e identificador y se comprenderá mucho mejor.

```
div p { color: red; }
```

Utilizar propiedades abreviadas

Siempre que sea posible es recomendable utilizar clases abreviadas para conseguir una reducción de código.

```
/* Propiedades margin-left, margin-right y margin-top */
.nav-menu {margin-left: 5px; margin-right: 5px; margin-top: 5px;}
/* Propiedad abreviada margin */
.nav-menu {margin: 5px 5px 0px 5px;}
```

Utilizar nombres descriptivos en los selectores

Mediante la utilización de nombres que permitan averiguar fácilmente a qué elemento le estamos dando estilo comprenderemos el código fácilmente.

```
.nav-button { background: blue; } /* Estilo del botón de la navbar*/
```

Evitar utilizar como nombre de un selector una característica visual

Al utilizar en el nombre de un selector una característica visual como el color, el tamaño o la posición, si posteriormente modificamos esa característica, también deberíamos cambiar el nombre del selector. Esto complica mucho el código ya que tendríamos que actualizar todas las referencias a ese selector en el código HTML.

```
/* Selector con nombre que define la característica visual del color */
.menu-red { background: red; }
/* Utilizar mejor: */
.nav-menu { background: red; }
```

Probar el diseño en los diferentes navegadores

Usar herramientas como: [ChromeFirefoxOperaSafariMicrosoft Edge](#)

También puedes usar la aplicación [browserling](#) que te permite ver tu desarrollo en varias versiones diferentes de cada navegador.



Validar el código CSS

Detectar errores en el código es esencial y se puede hacer fácilmente mediante un validador CSS. El W3C proporciona una [herramienta de validación de CSS](#) gratuita para los documentos CSS.

Agregar los prefijos de los navegadores en propiedades que no sean estables

Para ahorrar tiempo y facilitarnos la tarea de incluir los prefijos de las propiedades CSS que todavía no son estables podemos hacer uso de la [extensión «Autoprefixer» en Visual Studio Code](#).

5. PREFIJOS CSS DE LOS NAVEGADORES

Actualmente los navegadores tienen implementadas muchas de las nuevas características de CSS3 utilizando sus propias versiones de cada propiedad mediante prefijos.

Después de un tiempo, cuando las especificaciones son estables, se eliminarán las propiedades con prefijo. Los prefijos para los navegadores más comunes son los siguientes:

Prefijo	Navegador
-moz-	Firefox
-webkit-	Safari y Chrome
-o-	Opera
-khtml-	Konqueror
-ms-	Internet Explorer
-chrome-	Google Chrome

Así por ejemplo para transformar un elemento en Firefox, es necesario utilizar la propiedad `-moz-transform`; en los navegadores basados en WebKit, como Safari y Google Chrome, se utiliza la propiedad `-webkit-transform`. Por este motivo, no es de extrañar que en algunos casos, tengamos que añadir hasta cuatro líneas de código para una única propiedad CSS.

Se pueden ver los navegadores que soportan una determinada propiedad CSS o un elemento HTML5 en la página caniuse.com.

EJEMPLO <https://codepen.io/profe-ana/pen/RwyYPev>